

In the claims:

Following is a complete set of claims as amended with this Response.

1. (Currently Amended) A computer implemented method comprising:
receiving information regarding an ~~atomic distributed~~ a transaction at a
transaction coordinator, the ~~atomic distributed~~ transaction representing an
~~aggregation of a plurality of a discrete transactions~~ transaction for a
resource item ~~items that span a plurality of network resources~~;
sending from the transaction coordinator placing a tentative hold request for on
~~each of the plurality of resource item items by to a transaction manager~~
causing a tentative hold record to be created and associated with ~~each of~~
~~the plurality of the discrete transaction transactions~~, the tentative hold
holds operating in a non-mutually exclusive manner, ~~thereby allowing the~~
~~same resource item to be tentatively held by more than one transaction~~;
and
after successfully gaining the tentative hold on holds on each of the plurality of
the resource item items and receiving confirmation from the transaction
manager a confirmation regarding the ~~atomic distributed~~ transaction,
~~attempting to direct the transaction coordinator directing the completion~~
commitment of the ~~atomic distributed~~ transaction ~~by conventional means~~.
2. (Currently Amended) The method of claim 1, wherein said directing the
commitment of the transaction attempting to direct the completion of the atomic
~~distributed transaction by conventional means~~ comprises initiating conventional
Two-Phase Commit (2PC) prepare and commit processing for the transaction
~~each of the plurality of discrete transactions~~.

3. (Currently Amended) The method of claim 1, further comprising receiving a notification indicating the transaction ~~one of the plurality of discrete transactions~~ are no longer possible.
4. (Currently Amended) The method of claim 1, wherein ~~one or more of the~~ tentative hold record ~~records~~ are stored at an intermediate server that is not within the enterprise offering the resource item.
5. (Currently Amended) The method of claim 1, wherein the non-mutually exclusive manner of the tentative hold allows the resource item to be held for a short duration of time ~~the plurality of network resources comprise database systems of a plurality of different enterprises.~~
6. (Currently Amended) A computer implemented method comprising:
receiving information regarding a distributed transaction at a transaction coordinator from an originating application, the ~~distributed transaction~~ involving a resource item ~~plurality of items spanning a plurality of network resources;~~ and
the transaction coordinator initiating a tentative-hold processing stage by requesting that a resource manager ~~plurality of resource managers~~ residing on a remote server ~~one or more remote servers~~ and participating in the ~~distributed transaction~~ each by tentatively holding the item ~~hold an item of the plurality of items~~ involved in the ~~distributed transaction~~ and ~~storing~~ call back information identifying a return communication path to the originating application, ~~the~~ a tentative hold record ~~records~~ operating in a non-mutually exclusive manner, ~~thereby allowing items associated with the one or more remote servers to be tentatively held by more than one application.~~

7. (Currently Amended) The method of claim 6, wherein ~~at least two of the remote server is servers are~~ associated with different enterprises.
8. (Currently Amended) The method of claim 6, further comprising[:]
receiving a commitment corresponding to the ~~distributed~~ transaction from the
originating application; and
responsive to the commitment, initiating a two-phase commit processing stage by
the transaction coordinator directing the resource ~~manager managers~~ to
reserve the item items during which the resource ~~manager managers~~
reserve the item items and notifying, via corresponding call back
information, other applications having a tentative hold on the same items
that their respective tentative holds have been suspended.
9. (Currently Amended) A computer implemented method comprising:
receiving at a transaction coordinator, from a first client, a first request associated
with a first ~~discrete~~ transaction, the first request soliciting a non-mutually
exclusive hold on a resource item, ~~[[;]]~~ the resource item being part of the
first transaction ~~a first atomic distributed transaction that spans a plurality~~
~~of network resources~~;
maintaining a first non-mutually exclusive hold on the resource item until an
exclusive lock is obtained on the resource item or for a predetermined
amount of time, whichever occurs first, by the transaction coordinator
causing a first tentative hold record to be created and associated with the
resource item and initiating a first timeout associated with the first
tentative hold record;
receiving at the transaction coordinator, from a second client, a second request
associated with a second ~~discrete~~ transaction, the second request soliciting
a non-mutually exclusive hold on the resource item, the resource item

being part of the second transaction ~~a second atomic distributed transaction~~;

maintaining a second non-mutually exclusive hold on the resource item until an exclusive lock is obtained on the resource item or for a predetermined amount of time, whichever occurs first, by the transaction coordinator causing a second tentative hold record to be created and associated with the resource item and initiating a second timeout associated with the second tentative hold record;

receiving at the transaction coordinator, from the first client, a third request associated with the first ~~discrete~~ transaction, the third request asking that completion of the first ~~discrete~~ transaction commence; and responsive to the third request, the transaction coordinator suspending the second non-mutually exclusive hold and granting an exclusive lock on the resource item to the first ~~discrete~~ transaction.

10. (Currently Amended) The method of claim 9, wherein the first non-mutually exclusive hold allows the resource item to be held for a short duration of time at least two network resources of the plurality of network resources are associated with different enterprises.
11. (Currently Amended) The method of claim 9, further comprising:
storing call back information associated with an application originating the second ~~discrete~~ transaction; and
notifying the application regarding the suspension of the second non-mutually exclusive hold.

12. (Currently Amended) The method of claim 9, further comprising in response to a timeout on the exclusive lock, recommencing the second non-mutually exclusive hold on behalf of the second discrete transaction.
13. (Original) A distributed transaction processing system comprising:
a distributed transaction coordinator executing on a first client system, the distributed transaction coordinator to place non-mutually exclusive holds on each of a plurality of resource items associated with an atomic distributed transaction that spans a plurality of network resources and to commence completion of the atomic distributed transaction by obtaining exclusive locks on each of the plurality of resource items after non-mutually exclusive holds have been successfully granted on each of the plurality of resource items; and
a distributed transaction manager executing on a server system communicatively coupled with a plurality of client systems including the first client system, the distributed transaction manager to maintain a plurality of non-mutually exclusive holds for each of a plurality of resource items associated with the server system and to grant only one exclusive lock per single resource item of the plurality of resource items at a given time in response to requests from distributed transaction coordinators.
14. (Original) The distributed transaction processing system of claim 13, wherein the distributed transaction coordinator includes a Two-Phase Commit transaction coordinator.
15. (Original) The distributed transaction processing system of claim 13, further comprising one or more Two-Phase Commit resource managers communicatively coupled with the distributed transaction manager.

16. (Original) A machine-readable medium having stored thereon data representing sequences of instructions, the sequences of instructions which, when executed by a processor, cause the processor to:
- receive information regarding an atomic distributed transaction, the atomic distributed transaction representing an aggregation of a plurality of discrete transactions for individual resource items that span a plurality of network resources;
- place a tentative hold on each of the plurality of individual resource items by causing a tentative hold record to be created and associated with each of the plurality of discrete transactions, the tentative holds operating in a non-mutually exclusive manner, thereby allowing the same resource item to be tentatively held by more than one interested party; and
- after successfully gaining the tentative holds on each of the plurality of individual resource items and receiving a confirmation regarding the atomic distributed transaction, attempt to direct the completion of the atomic distributed transaction by conventional means.
17. (Original) The machine-readable medium of claim 16, wherein said attempt to direct the completion of the atomic distributed transaction by conventional means comprises initiating conventional Two-Phase Commit (2PC) prepare and commit processing for each of the plurality of discrete transactions.
18. (Original) The machine-readable medium of claim 16, wherein one or more of the tentative hold records are stored at an intermediate server that is not within the enterprise offering the resource item.
19. (Original) The machine-readable medium of claim 16, wherein the plurality of network resources comprise database systems of a plurality of different enterprises.

20. (Original) A machine-readable medium having stored thereon data representing sequences of instructions, the sequences of instructions which, when executed by a processor, cause the processor to:
- receive, from a first client, a first request associated with a first discrete transaction, the first request soliciting a non-mutually exclusive hold on a resource item; the resource item being part of a first atomic distributed transaction that spans a plurality of network resources;
- maintain a first non-mutually exclusive hold on the resource item until an exclusive lock is obtained on the resource item or for a predetermined amount of time, whichever occurs first, by causing a first tentative hold record to be created and associated with the resource item and initiating a first timeout associated with the first tentative hold record;
- receive, from a second client, a second request associated with a second discrete transaction, the second request soliciting a non-mutually exclusive hold on the resource item, the resource item being part of a second atomic distributed transaction;
- maintain a second non-mutually exclusive hold on the resource item until an exclusive lock is obtained on the resource item or for a predetermined amount of time, whichever occurs first, by causing a second tentative hold record to be created and associated with the resource item and initiating a second timeout associated with the second tentative hold record;
- receive, from the first client, a third request associated with the first discrete transaction, the third request asking that completion of the first discrete transaction commence; and
- responsive to the third request, suspend the second non-mutually exclusive hold and grant an exclusive lock on the resource item to the first discrete transaction.

21. (Original) The machine-readable medium of claim 20, wherein at least two network resources of the plurality of network resources are associated with different enterprises.
22. (Original) The machine-readable medium of claim 20, wherein the sequences of instructions further include instructions which, when executed by the processor, cause the processor to:
store call back information associated with an application originating the second discrete transaction; and
notify the application regarding the suspension of the second non-mutually exclusive hold.
23. (Currently Amended) The machine-readable medium method of claim 20, wherein the sequences of instructions further include instructions which, when executed by the processor, cause the processor to recommence the second non-mutually exclusive hold on behalf of the second discrete transaction in response to a timeout on the exclusive lock.
24. (New) The method of claim 6, wherein the non-mutually exclusive manner of the tentative hold allows the resource item to be held for a short duration of time.